Finite-Trace and Generalized-Reactivity Specifications in Temporal Synthesis

Giuseppe De Giacomo[1], Antonio Di Stasio[1], Lucas M. Tabajara[2], Moshe Vardi[2] and **Shufang Zhu**[1]

[1]Sapienza University of Rome
[2]Rice University

Introduction

- Temporal Synthesis: automatically design a reactive system with correctness guarantee [Pnueli&Rosner, 1989]

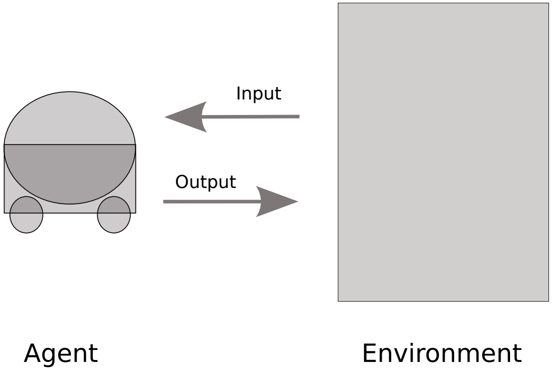


Figure 1: Reactive System

Temporal Synthesis: Challenges and Successes

- Linear Temporal Logic (LTL), games on infinite-word automata
  - Parity game construction and solving
  - No scalable algorithms [Kupferman, 2012]
- Two successful responses to the difficulties
  - Developed in Formal Methods, GR(1) approach [Bloem et al., 2012]
  - Developed in AI, LTL_f approach [De Giacomo&Vardi, 2013]
  - Numerous applications, e.g., planning [Kress-Gazit et al., 2009; Camacho et al., 2017; De Giacomo&Rubin, 2018]

What We Want

- Bring two successes together
  - LTL_f approach
  - GR(1) approach
- Applicable scenarios
  - Planning, nondeterministic planning domain

Generalized Reactivity (1), GR(1) [Bloem et al., 2012]

- GR(1) game $\mathcal{G} = (\mathcal{A}, \psi)$
  - Game arena $\mathcal{A}$ encoding moves
  - Winning condition GR(1) formula ψ, powerful notion of fairness

$$\psi = \bigwedge_{i=1}^{m} \Box\Diamond\mathcal{J}_i \rightarrow \bigwedge_{j=1}^{n} \Box\Diamond\mathcal{K}_j$$

- **Pros**: simple game solving, quadratic time
- **Cons**: only fairness, weak for specifying agent tasks

LTL over finite traces, LTL_f [De Giacomo&Vardi, 2015]

- Accomplishing task in LTL_f iff DFA game with reachability condition
- **Pros**: Applicable with simple solution
  - Finite-horizon agent tasks, AI applications
  - Simple game arena construction, finite-word automata
- **Cons**: Having environment assumptions $\varphi^e \rightarrow \varphi^a_{task}$
  - Agent task φ^a_{task} in LTL_f
  - Environment assumption φ^e in LTL
  - Going back to LTL synthesis?

LTL_f Synthesis under Env. Assumptions $\varphi^e \rightarrow \varphi^a_{task}$

- LTL, safe LTL, co-safe LTL [Camacho et al., 2018; De Giacomo et al., 2020]
  - Still infinite-word automata, not scalable
- restriction on the form of the specifications themselves Simple Fairness (*infinitely-often*) and Stability (*eventually-always*) [Zhu et al., 2020]
  - Game arena from DFA of LTL_f φ^a_{task}
  - φ^e as part of the DFA game winning condition
  - Limited expressiveness, $\varphi^e = \Box\Diamond J$ or $\varphi^e = \Diamond\Box K$

Bring Together GR(1) and LTL_f

- GR(1) formula, generalization of fairness
  - Powerful expressiveness on environment assumption
- LTL_f formula, natural for finite-horizon task
  - Strong agent task specification
- Simple solution
  - No detour to LTL synthesis
  - Games on finite-word automata

LTL_f under GR(1) Assumptions

Given:

- Environment variables $\mathcal{X}$, Agent variables $\mathcal{Y}$
- Agent task φ^a_{task} in LTL_f
- Environment assumption $\varphi^e_{GR(1)} = \bigwedge_{i=1}^{m} \Box\Diamond\mathcal{J}_i \rightarrow \bigwedge_{j=1}^{n} \Box\Diamond\mathcal{K}_j$

Obtain:

Agent strategy $g : (2^{\mathcal{X}})^* \rightarrow 2^{\mathcal{Y}}$, a function from past history of environment behaviors to agent actions

- **If**: environment behaves as $\varphi^e_{GR(1)}$
- **Then**: φ^a_{task} is achieved

Reduction to Two-player Games

- Agent goal: **IF** $\varphi^{e}_{GR(1)}$ **THEN** φ^{a}_{task}
- **Key Idea:** Conduct specific two-player games
  - LTL_f task, corresponding DFA as the game arena
  - GR(1) assumption, part of the game winning condition

The agent wins the game if a specific winning condition is satisfied.

Reduction to Two-player Games

- Agent goal: **IF** $\varphi^e_{GR(1)}$ **THEN** φ^a_{task}
- Game arena from DFA of φ^a_{task}
- The agent wins the game if either happens:
  - Reachability: visiting the final states from φ^a_{task}
  - Dual GR(1): violating $\varphi^e_{GR(1)}$
- Complexity: quadratic time

Applicable Scenarios: Planning

- Planning domain, safety conditions on both players
- Adding safety conditions
  - *Environment safety:* φ^e_{safe} holds forever
  - *Agent safety:* φ^a_{safe} holds until φ^a_{task} is accomplished
  - **IF** $\varphi^e_{GR(1)} \wedge \varphi^e_{safe}$ **THEN** $\varphi^a_{task} \wedge \varphi^a_{safe}$

Safety Conditions

- Safety: bad things never happen
    - Infinite-word automata? Not necessary!
- Alternative way: interpretation on all prefixes
    - No bad prefixes, all prefixes are good
    - Prefixes are finite

Safety Conditions in LTL_f

- LTL_f formula φ
- Satisfaction on all prefixes: a (finite or infinite) trace π satisfies φ
  - Every non-empty finite prefix of π satisfies φ
- Capture all safety properties! Finite and Infinite!

Reduction to Two-player Games

- Agent goal: **IF** $\varphi^e_{GR(1)} \wedge \varphi^e_{safe}$ **THEN** $\varphi^a_{task} \wedge \varphi^a_{safe}$
- Alternative view: **IF** $\varphi^e_{GR(1)}$ **THEN** $(\sim \varphi^e_{safe}) \vee (\varphi^a_{task} \wedge \varphi^a_{safe})$
  - Game arena from DFAs of $\varphi^e_{safe}, \varphi^a_{task}, \varphi^a_{safe}$
  - The agent wins the game if any happens:
    - Reachability: visiting non-final states from φ^e_{safe}
    - Reachability: visiting final states from φ^a_{task} while staying in final states from φ^a_{safe}
    - Dual GR(1): Violating $\varphi^e_{GR(1)}$
  - Complexity: quadratic time

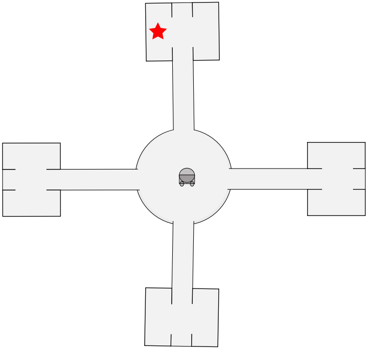
LTL_f under GR(1) Assumptions with Safety Conditions

- Problem: *if* $\varphi^e_{GR(1)} \wedge \varphi^e_{safe}$ *then* $\varphi^a_{task} \wedge \varphi^a_{safe}$
- Game arena construction, simple
  - LTL_f formulas $\varphi^e_{safe}, \varphi^a_{safe}, \varphi^a_{task}$, finite-word automata
- Game solving, simple
  - Quadratic time

Experimental Evaluation

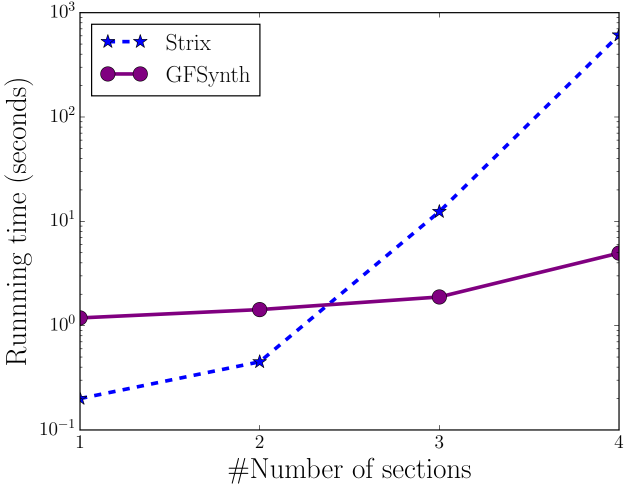
- GFSynth: automaton construction, reduction to GR(1) game, and game solving
  - LTL_f-to-DFA Syft [Zhu et al., 2017]
  - GR(1) game solver Slugs [Ehlers&Raman, 2016]
- Baseline: reduction to LTL synthesis, Strix [Meyer et al., 2018]
- Benchmarks: Finding Nemo, based on [Kress-Gazitet al., 2009]

Finding Nemo



- Circular hallway leading to n sections, two rooms of each
- Robot looking for "Nemo", appear in any of the odd-numbered rooms
- **Agn. Task**: 3 times of Nemo
- **Env. GR(1)**: $(\Box\Diamond$ Visit_Odd-numbered-rooms$) \rightarrow (\Box\Diamond$ Nemo_Appears$)$

Experimental Results: Finding Nemo



LTL_f under GR(1) Assumptions with Safety Conditions

- Problem: *if* $\varphi^e_{GR(1)} \wedge \varphi^e_{safe}$ *then* $\varphi^a_{task} \wedge \varphi^a_{safe}$
- Writing safety conditions in LTL_f
- Simple game arena construction
  - DFA construction from LTL_f formulas $\varphi^e_{safe}, \varphi^a_{safe}, \varphi^a_{task}$
- Simple game solving
  - GR(1) formula $\varphi^e_{GR(1)}$